

**CLAIMS**

What is claimed is:

- 5     1. In an electronic device, a cross-platform interface tool for automatically creating a common programming interface for a plurality of components, comprising:  
          a parsing mechanism for parsing through a component description associated with at least one of the plurality of components to gather information relating to the at least one of the plurality of components; and  
10           a component interface creator for automatically generating a component interface based on the information gathered by the parsing mechanism.
- 15     2. The cross-platform interface tool of claim 1, wherein the component description includes information relating to component features and interface requirements.
3. The cross-platform interface tool of claim 1, wherein the component description comprises an advertising mechanism for advertising the information.
4. The cross-platform interface tool of claim 3, wherein the advertising mechanism  
20     comprises data.
5. The cross-platform interface tool of claim 3, wherein the advertising mechanism comprises a software object definition.
- 25     6. The cross-platform interface tool of claim 3, wherein the component description comprises a communications system.
7. The cross-platform interface tool of claim 1, wherein the component interface creator  
30     comprises a lookup mechanism for searching existing component interfaces to determine whether a previously established component interface already exists.

8. The cross-platform interface tool of claim 1, wherein the component interface creator comprises a class generator for generating a class based on the information gathered by the parsing mechanism if there is no pre-existing component interface.
- 5     9. The cross-platform interface tool of claim 1, wherein the component interface creator comprises an instantiator for instantiating a class generated by a class generator to create a device object for interfacing with the at least one of the plurality of components if there is no pre-existing component interface.
- 10    10. The cross-platform interface tool of claim 1, wherein the component comprises at least one of a hardware device, a data or image acquisition device, and an embedded device.
11. The cross-platform interface tool of claim 1, further comprising a testing tool for  
15    verifying performance of the component interface.
12. The cross-platform interface tool of claim 11, wherein a test originating with the testing tool can be saved in file format by the cross-platform interface tool.
13. The cross-platform interface tool of claim 12, wherein the file format comprises a  
20    driver format.
14. In an electronic device, a method of automatically creating a common programming interface for a plurality of components using a cross-platform interface tool, the method  
25    comprising:
- parsing through a component description associated with at least one of the plurality of components to gather information relating to the at least one of the plurality of components; and
- automatically generating a component interface based on the information  
30    gathered by the parsing mechanism.
15. The method of claim 14, wherein the component description includes information relating to component features and interface requirements.

16. The method of claim 14, wherein the component description comprises an advertising mechanism advertising the information.
- 5 17. The method of claim 16, wherein the advertising mechanism comprises data.
18. The method of claim 16, wherein the advertising mechanism comprises a software object definition.
- 10 19. The method of claim 16, wherein the component description comprises a communications system.
20. The method of claim 14, wherein the step of automatically generating a component interface comprises a lookup mechanism searching existing component interfaces to  
15 determine whether a previously established component interface already exists.
21. The method of claim 14, wherein the step of automatically generating a component interface comprises a class generator generating a class based on the information gathered by the parsing mechanism if there is no pre-existing component interface.  
20
22. The method of claim 14, wherein the step of automatically generating a component interface comprises an instantiator instantiating a class generated by a class generator to create a device object for interfacing with the at least one of the plurality of components if there is no pre-existing component interface.  
25
23. The method of claim 14, wherein the component comprises at least one of a hardware device, a data or image acquisition device, and an embedded device.
24. The method of claim 14, further comprising verifying performance of the  
30 component interface utilizing a testing tool.
25. The method of claim 24, further comprising saving a test originating with the testing tool in file format.

26. The method of claim 25, wherein the file format comprises a driver format.
27. A medium for use in a modeling and execution environment on an electronic device,  
5 the medium holding instructions executable using the electronic device for performing a method of automatically creating a common programming interface for a plurality of component using a cross-platform interface tool, the method comprising:  
parsing through a component description associated with at least one of the  
plurality of components to gather information relating to the at least one of the plurality  
10 of components; and  
automatically generating a component interface based on the information gathered by the parsing mechanism.
28. The medium of claim 27, wherein the component description includes information  
15 relating to component features and interface requirements.
29. The medium of claim 27, wherein the component description comprises an advertising mechanism advertising the information.
- 20 30. The medium of claim 29, wherein the advertising mechanism comprises data.
31. The medium of claim 29, wherein the advertising mechanism comprises a software object definition.
- 25 32. The medium of claim 29, wherein the component description comprises a communications system.
33. The medium of claim 27, wherein the step of automatically generating a component  
interface comprises a lookup mechanism searching existing component interfaces to  
30 determine whether a previously established component interface already exists.

34. The medium of claim 27, wherein the step of automatically generating a component interface comprises a class generator generating a class based on the information gathered by the parsing mechanism if there is no pre-existing component interface.
- 5     35. The medium of claim 27, wherein the step of automatically generating a component interface comprises an instantiator instantiating a class generated by a class generator to create a device object for interfacing with the at least one of the plurality of components if there is no pre-existing component interface.
- 10    36. The medium of claim 27, wherein the component comprises at least one of a hardware device, a data or image acquisition device, and an embedded device.
37. The medium of claim 27, further comprising a testing tool for verifying performance of the component interface.
- 15    38. The medium of claim 37, further comprising saving a test originating with the testing tool in file format.
39. The medium of claim 38, wherein the file format comprises a driver format.
- 20    40. In an electronic device running a modeling software application, a cross-platform interface tool for automatically creating a common programming interface for a plurality of hardware instruments, comprising:
- 25        a parsing mechanism for parsing through a hardware instrument description associated with at least one of the plurality of hardware instruments to gather information relating to the at least one of the plurality of hardware instruments; and
- a hardware instrument interface creator for automatically generating a hardware instrument interface based on the information gathered by the parsing mechanism.
- 30    41. The cross-platform interface tool of claim 40, wherein the hardware instrument description includes information relating to hardware instrument features and interface requirements.

42. The cross-platform interface tool of claim 40, wherein the hardware instrument description comprises an advertising mechanism for advertising the information.
43. The cross-platform interface tool of claim 42, wherein the advertising mechanism  
5 comprises data.
44. The cross-platform interface tool of claim 42, wherein the advertising mechanism comprises a software object definition.
- 10 45. The cross-platform interface tool of claim 42, wherein the hardware instrument description comprises a communications system.
46. The cross-platform interface tool of claim 40, wherein the hardware instrument interface creator comprises a lookup mechanism for searching existing hardware  
15 instrument interfaces to determine whether a previously established hardware instrument interface already exists.
47. The cross-platform interface tool of claim 40, wherein the hardware instrument interface creator comprises a class generator for generating a class based on the  
20 information gathered by the parsing mechanism if there is no pre-existing hardware instrument interface.
48. The cross-platform interface tool of claim 40, wherein the hardware instrument interface creator comprises an instantiator for instantiating a class generated by a class  
25 generator to create a device object for interfacing with the at least one of the plurality of hardware instruments if there is no pre-existing hardware instrument interface.
49. The cross-platform interface tool of claim 40, further comprising a testing tool for verifying performance of the hardware instrument interface.  
30
50. The cross-platform interface tool of claim 49, wherein a test originating with the testing tool can be saved in file format by the cross-platform interface tool.

51. The cross-platform interface tool of claim 50, wherein the file format comprises a driver format.